UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 09/595,003 | 06/13/2000 | Nicolas Vazquez | 5150-44300 | 7955 |

7590 03/22/2007

Jeffrey C. Hood
Meyertons, Hood, Kivlin, Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767

| EXAMINER | |
|---|---|
| PILLAI, NAMITHA | |
| **ART UNIT** | **PAPER NUMBER** |
| 2173 | |

| SHORTENED STATUTORY PERIOD OF RESPONSE | MAIL DATE | DELIVERY MODE |
|---|---|---|
| 3 MONTHS | 03/22/2007 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

| Office Action Summary | Application No. | Applicant(s) | |
|---|---|---|---|
| | 09/595,003 | VAZQUEZ ET AL. | |
| | Examiner | Art Unit | |
| | Namitha Pillai | 2173 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *08 December 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-7,9-37,39-59,61-74 and 76-90* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-7,9-37,39-59,61-74 and 76-90* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *1/10/07*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

## DETAILED ACTION

### *Response to Amendment*

1. The Examiner acknowledges Applicant's submission on 12/8/06 including amendments to claim 45 and arguments against the previous rejection. Applicant's arguments related to EP Publication No. 0510514 A1 (Oka et al.) not disclosing automatically generating a program is persuasive, but all pending claims have been rejected as being obvious over the prior arts disclosed in the field of graphical program generation.

### *Claim Rejections - 35 USC § 103*

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-16, 21-37, 39-43, 45-59, 61-65, 67-74 and 76-90 are rejected under 35 U.S.C. 103(a) as being unpatentable over U. S. Patent No. 5,862,372 (Morris et al.), herein referred to as Morris, EP Publication No. 0510514 A1 (Oka et al.), herein referred to as Oka and U. S. Patent No. 4,831,580 (Yamada).

Referring to claims 1, 53, 71, 81 and 90, Morris discloses a method for creating a graphical program to perform an algorithm, with the recording of functions in response to user input, with the functions specifying the algorithm (column 2, lines 20-27). Morris discloses automatically generating the graphical program in response to the recorded functions, with the display of a plurality of interconnected nodes, which visually indicate

functionality of the graphical program, with the graphical program implementing the algorithm. See column 3, lines 29-35. Morris does not disclose that user input does not use the selection of the nodes. Oka discloses automatically generating a graphical program represented as the flowchart with the program functions, where the blocks or nodes representing each function is automatically generated, selected and formed into a graphical program without any user intervention with the graphical program being automatically generated (column 1, lines 20-26). It would have been obvious for one skilled in the art, at the time of the invention to learn from Oka to implement an automatic process in which both the blocks and the connections between the blocks are automatically generated to implement a program. The systems and objectives of creating a graphical program are similar in both Morris and Oka. Both inventions describe a process for generating a graphical program, with the nodes of the program representing functions. Oka alleviates user interaction by further teaching automatically generating of the graphical flowchart without any user intervention. The graphical program of Oka is generated by automatically generating the blocks or nodes representing the functions and automatically connecting the nodes to generate a representation of the graphical program. Hence, one skilled in the art would have been motivated to learn from Oka to automatically generate the flowchart, through including of the nodes without direct user input selecting the nodes.

Morris and Oka do not disclose that the automatically created flowchart is generated based on an algorithm, where the flowchart generation involves creation of a program. Yamada discloses a flowchart which when created, generates a program

(column 2, lines 8-27). It would have been obvious to one skilled in the art at the time of the invention to learn from Yamada that the flowchart generated is a program that is created for execution from the generated flowchart. The combination of Morris and Oka have taught that a flowchart can automatically be generated with Yamada further teaching the generated flowcharts can represent program that are generated when the flowchart is created for automatic generation of a program. The combination of Morris and Oka could further automate a previously manual process by relying on Yamada to learn that an automatically generated flowchart can be used in automatic creation of a graphical program. Therefore, one skilled in the art would have been motivated to learn from Yamada to that the flowchart generated creates a program automatically.

Referring to claims 2, 54, 79 and 88, Morris discloses performing the function in response to user input, wherein the user dragging the objects is the function which is then recorded in response to the user's action for specifying the algorithm (column 3, lines 32-34).

Referring to claims 3, 32 and 55, Morris discloses creating a prototype or a set of instructions by recording the functions carried out by the user (column 3, lines 29-30).

Referring to claims 4, 33, 56, 72 and 82, Morris discloses that the prototype is in the discipline of image processing as seen by the results of the prototyping program shown in Figure 5 (column 6, lines 37-42).

Referring to claims 5, 34 and 57, Morris discloses recording the functions in response to input received via a graphical user interface (column 2, lines 20-24 and Figure 5).

Referring to claims 6, 58, 73 and 83, Morris discloses that the graphical user interface wherein the user would work with is based on the prototyping environment application (column 2, lines 57-52).

Referring to claims 7, 35, 59, 74 and 84, Morris discloses that the user input consists of selecting the functions from a menu and palette (column 2, lines 20-24).

Referring to claims 9, 39, 61, 76 and 85, Oka discloses automatically generating the graphical program comprises automatically including and connecting the nodes generating graphical code in the graphical program without direct user input (column 1, lines 20-26).

Referring to claims 10 and 62, Morris discloses the graphical program running or executing, wherein the algorithm represented by the functions are performed (column 3, lines 34-35).

Referring to claims 11, 40, 63 and 87, Morris discloses that the graphical program includes a block diagram portion (column 5, line 43) and a user interface panel portion, represented as the palette in Figure 5.

Referring to claims 12, 41 and 78, Morris discloses that the graphical program is a graphical data flow program as seen in Figure 2 (column 5, lines 43-46).

Referring to claims 13, 42, 64, 77 and 86, Oka discloses that the automatic generation of the graphical program is done through the automatic inclusion of nodes or "objects" corresponding to respective one of the one or more functions in the graphical program (column 1, lines 20-26).

Referring to claims 14, 43 and 65, Morris discloses the functions comprising a script, with the script having an association with the graphical program (column 6, lines 24-26). Morris discloses modifying the script to create a new script in response to user input once an association has been made (column 6, lines 26-33). Morris also discloses modifying the graphical program, shown as the map view of objects for the graphical program, and this map representation being modified based on the changes to the script, with the production of the new script (column 6, lines 32-43).

Referring to claim 15, Morris discloses there being a clear association between the script and the graphical program, this association being used during modification of the graphical program and this association remaining between the new script and the new graphical program (column 5, lines 54-55 and column 6, lines 24-36).

Referring to claim 16, Morris discloses receiving user input indicating a desire to change the graphical program, displaying the script information of the script, modifying the script information in response to user input and modifying the graphical program after modifying the script information. See column 6, lines 9-14).

Referring to claims 21, 45 and 67, Morris discloses receiving user input specifying code generation information and using this information to automatically generate the graphical program (column 2, lines 20-27).

Referring to claims 22, 46 and 68, Morris discloses that the code generation information represented as objects, specify or represent the type of program to create in response to the recorded function, wherein the program is created in accordance with the specified graphical program type (column 3, lines 5-15).

Referring to claims 23 and 47, Morris does disclose that the graphical program type, in this case being "WINDOWS" applications, specifies a particular programming environment, wherein the program, represented as "APPLICATIONS" are created in a file format that is usable by the particular programming environment. Morris clearly discloses using standard programming languages, representing the file format, which would enable the users to create the applications particular to the programming environment being used. See column 4, lines 45-55.

Referring to claims 24, 48 and 69, Morris discloses a plurality of parameters associated with the functions wherein each parameter is an input parameter, which provides input to a function (column 6, lines 26-29). Morris also discloses that the code generation information specifies input parameters, which are desired to be interactively changeable. Morris also discloses automatically generating the graphical program represented by objects, wherein the program receives user input during the program operation, with the user specifying values for the specified input parameters. See column 6, lines 24-32. Morris also discloses automatically generating the graphical program comprises enabling the graphical program to display output during program operation, wherein the output indicates values for the specified parameters (column 6, lines 37-41).

Referring to claims 25, 49, 70, 80 and 89, Morris discloses automatically generating a graphical program includes generating portions of graphical code, with each portion implementing one of the functions and linking the portions of graphical code together (column 1, lines 61-66 and column 2, lines 1-4).

Referring to claims 26 and 50, as seen in Figure 2 of Morris, the graphical programming nodes each have inputs and outputs, and wherein generating the portions of the graphical code comprises connecting the node inputs and outputs together in order to implement the function with which the portion of graphical code is associated (column 5, lines 40-50).

Referring to claims 27 and 51, as seen in Figure 2 of Morris, a first portion of graphical code is linked to a second portion of graphical by connecting an output of a node in the first portion to an input of a node in the second portion of the graphical code.

Referring to claim 28, Morris discloses data being passed between the objects, wherein the data affecting the outcome of these objects (column 5, lines 46-48). As seen in Figure 2 also, the functions represented as the objects have input parameters, wherein the portion of code with the node has an input for receiving a value for the input parameter. Each of the nodes having an input parameter also has a leaf node that has an output for providing a value for the input parameter, with the leaf node for providing the parameter value is connected to the node input for receiving the parameter value as seen by the node relationships shown in Figure 2.

Referring to claim 29, Morris discloses that the functions have output parameters, with as seen in Figure 2, certain nodes providing output parameters to other node, to implement functions, wherein there is a leaf node that is associated with the node with the output parameter, with the leaf node receiving the output parameter as input for the node, with the two nodes being connected to each other. See Figure 2.

Referring to claims 30 and 52, Morris discloses that al l information used for this invention, which would include the information needed to generate the graphical program and the functionalities of the nodes are all stored remote information source or storage means, thereby suggesting a database, as shown in Figure 1 (column 5, lines 10-16).

Referring to claim 31, Morris discloses a computer system, which would include a processor with a memory, coupled to the processor, which would store certain applications, one of them being a prototyping environment application (column 5, lines 8-11). Morris also discloses a user input device which receives user input, the presence of a user input obviously being inherently disclosed through the discussion of the user manipulating the icons from the palette (column 2, line 24). Morris discloses that this prototyping environment application carries out the generation of the graphical program (column 1, lines 10-15). Morris discloses a method for creating a graphical program to perform an algorithm, with the recording of functions in response to user input, with the functions specifying the algorithm (column 2, lines 20-27). Morris discloses automatically generating the graphical program in response to the recorded functions, with the display of a plurality of interconnected nodes, which visually indicate functionality of the graphical program, with the graphical program implementing the algorithm. See column 3, lines 29-35. Morris does not disclose that user input does not use the selection of the nodes. Oka discloses automatically generating a graphical program represented as the flowchart with the program functions, where the blocks or nodes representing each function is automatically generated, selected and formed into

a graphical program without any user intervention with the graphical program being automatically generated (column 1, lines 20-26). It would have been obvious for one skilled in the art, at the time of the invention to learn from Oka to implement an automatic process in which both the blocks and the connections between the blocks are automatically generated to implement a program. The systems and objectives of creating a graphical program are similar in both Morris and Oka. Both inventions describe a process for generating a graphical program, with the nodes of the program representing functions. Oka alleviates user interaction by further teaching automatically generating of the graphical flowchart without any user intervention. The graphical flowchart of Oka is generated by automatically generating the blocks or nodes representing the functions and automatically connecting the nodes to create a flowchart. Hence, one skilled in the art would have been motivated to learn from Oka to automatically generate the flowchart, through including of the nodes without direct user input selecting the nodes.

Morris and Oka do not disclose that the automatically created flowchart is generated based on an algorithm, where the flowchart generation involves creation of a program. Yamada discloses a flowchart which when created, generates a program (column 2, lines 8-27). It would have been obvious to one skilled in the art at the time of the invention to learn from Yamada that the flowchart generated is a program that is created for execution from the generated flowchart. The combination of Morris and Oka have taught that a flowchart can automatically be generated with Yamada further teaching the generated flowcharts can represent program that are generated when the

flowchart is created for automatic generation of a program. The combination of Morris and Oka could further automate a previously manual process by relying on Yamada to learn that an automatically generated flowchart can be used in automatic creation of a graphical program. Therefore, one skilled in the art would have been motivated to learn from Yamada to that the flowchart generated creates a program automatically.

Referring to claim 36, Morris discloses a computer system implementing this graphical program creation program, wherein the information would be stored in memory and there would be means for calling the graphical program creation program, within the computer system, as would be the case for calling any application in a computer system (column 5, lines 8-13). Morris also discloses that the prototyping environment application through calling the run time program, whereby calling the graphical program creation program, executes to automatically generate the graphical program (column 7, lines 4-7).

Referring to claim 37, Morris discloses that the graphical creation program is a graphical programming development environment application (column 4, lines 46-50).

3. Claims 17-20, 44 and 66 are rejected under 35 U.S.C. 103(a) as being unpatentable over Morris, Oka and Yamada, and further in view of U. S. Patent No. 5,623,659 (Shi et al.).

Referring to claims 17, 44 and 66, Morris and Oka discloses creating an association between the script and the graphical program (Morris, column 6, lines 30-32). Morris, Oka and Yamada do not disclose locking the association between the script and the general program. Shi discloses locking the association between a

program and a user, wherein the locking prevents other users from editing the portion associated to the initial user (column 2, lines 5-11). It would have been obvious to modify Morris, Oka and Yamada's invention such that locking the association between the script and the graphical program wherein the locking prevents the user from editing the program. Morris discloses allowing the user's to manipulate the scripts and the graphical program, but does not disclose any means for controlling this manipulation, thereby possibly leading to mistakes. Thus measures must be taken to ensure that unnecessary mistakes do not occur. The locking mechanism disclosed by Shi could prevent such mistakes from occurring in Morris, Oka and Yamada's graphical program, and laying out some control mechanisms for user manipulation. One skilled in the art would be motivated to learn from Shi's teachings by locking the association between the script and the graphical program to have more control over the user's manipulations of the graphical program.

Referring to claim 18, Shi discloses unlocking the association between the script and the graphical program in response to user input after locking. Shi also discloses directly changing the portion in response to the changes made by the user. See column 2, lines 19-23.

Referring to claim 19, Shi discloses unlocking such that it removes the association between the script and the program (column 2, lines 19-20).

Referring to claim 20, Morris discloses modifying the graphical program in response to user input after generating the graphical program and after creating the association between the script and the graphical program (column 6, lines 20-36).

Morris also discloses determining if an association exists between the script and the graphical program in response to the modifications made by the user concerning components of the graphical program (column 6, lines 30-32). Morris, Oka and Yamada do not disclose removing the association between the script and the graphical program. Shi discloses removing the association between the script and the graphical program in response to modifying (column 2, lines 19-20). It would have been obvious for one skilled in the art, at the time of the invention to remove the association between the script and the graphical program in response to modifying. Morris, Oka and Yamada's invention denotes an association between the script and the graphical program at all times, which could prove inconvenient, especially when the user may make mistakes by changing the script and the association causing the same changes to the graphical program which may be undesirable. To prevent such unnecessary mistakes, Morris, Oka and Yamada could use a mechanism for removing the association between the script and the graphical program in response to modifying. One skilled in the art, at the time of the invention, would have been motivated to learn from Shi to implement a means for removing the association between the script and the graphical program in response to modifying.

***Response to Arguments***

4. Applicant's arguments, filed 12/8/06, with respect to Oka not disclosing generating and creating a program have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of Yamada.

Oka has clearly disclosed the inconvenience of relying on user input to create a graphical structure that would represent the program and has disclosed in Oka how automatic generation of such a structure representing generation of a program can be convenient. Therefore, motivation has been provided for the use of such automatic generation of a structure, where functions representing blocks are automatically laid out and automatically connected to generate a flowchart. The flowchart of Oka is based on a predetermined program, where the flowchart is generated based on the program. Yamada has disclosed a flowchart, which is generated to create the program, where this flowchart upon creation automatically generates a program.

5. Applicant's arguments filed 12/8/06 with respect to the dependent claims have been fully considered but they are not persuasive.

Applicant argues that for claims 2 and 54, the functions to be recorded is defined as a specific type of function used in explaining in the arguments but not clearly stated within the claims. Function can be a set of steps taken to carry out a process, thereby the user dragging and dropping is interpreted as a processing function carried out in association with the program. Furthermore, the node that is dragged and dropped represents a function, where in response to the user input of drag and drop, the functions are recorded or set as part of the graphical program that is being created.

In reference to claims 11, 40, and 63, Figure 5 of Morris clearly discloses a map view window showing a block diagram, the block diagram is included in the graphical portion and a user interface panel portion can be interpreted as any of the panels that represent the nodes or functions within the block diagram. The panels of Morris also

are associated with the graphical program, where all the views within each panel represent the graphical program. Within one view of the Map View Window, where the block diagram is displayed, there is also a Palette, which represents a user interface panel portion.

With respect to claims 22, 23, 46, 47 and 68, with the graphical program being created to generate code, there is an association between the operations chosen by the user and the type of graphical program that will be created. Based on a user's choice, a distinct format will be associated with the generated graphical program with the code that is generated being associated with this graphical program. As long as the graphical programs generated have varying formats or layouts, there is represented graphical programs of different types. Although the program maybe the same, there are different types of programs, where the type of program reads on a program that belongs in a distinct type or category, which may be based on format, layout or the programming language that may be used.

With respect to claims 24, 48, and 69, program operation is not necessarily narrowed down to the period of runtime of the program. Program operation can represent, a process or action related to a program, which reads on the program being operated on in various ways including configuration or processing and also including execution. Therefore, Morris discloses specifying parameters and other modifications during program operation.

With respect to claim 17, regardless of Shi discloses locking data sets, where the data sets can represent various types of data including an association between a script

and a graphical program. Furthermore, Shi points out that using the lock method to keep up with versions of one distinct system component can attain version control. With Morris teaching modifying of the script in various conditions, in order to attain version control, a lock method must be implemented to keep up with various version of the script that be created.

***Conclusion***

6. Responses to this action should be submitted as per the options cited below: The United States Patent and Trademark Office requires most patent related correspondence to be: a) faxed to the Central Fax number (571-273-8300) b) hand carried or delivered to the Customer Service Window (located at the Randolph Building, 401 Dulany Street, Alexandria, VA 22314), c) mailed to the mailing address set forth in 37 CFR 1.1 (e.g., P.O. Box 1450, Alexandria, VA 22313-1450), or d) transmitted to the Office using the Office's Electronic Filing System.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Namitha Pillai whose telephone number is (571) 272-4054. The examiner can normally be reached on 8:30 AM - 5:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kristine Kincaid can be reached on (571) 272-4063.

All Internet e-mail communications will be made of record in the application file. PTO employees do not engage in Internet communications where there exists a possibility that sensitive information could be identified or exchanged unless the record includes a properly signed express waiver of the confidentiality requirements of 35

U.S.C. 122. This is more clearly set forth in the Interim Internet Usage Policy published in the Official Gazette of the Patent and Trademark on February 25, 1997 at 1195 OG 89.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the Group receptionist whose telephone number is (571) 272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Namitha Pillai
Assistant Examiner
Art Unit 2173
March 19, 2007

RAYMOND J. BAYERL
PRIMARY EXAMINER
ART UNIT 2173